Network Working Group
Request for Comments: 971

Annette L. DeSchon
ISI
January 1986

A SURVEY OF DATA REPRESENTATION STANDARDS

Status of This Memo

This RFC discusses data representation conventions in the ARPA-Internet and suggests possible resolutions. No proposals in this document are intended as standards for the ARPA-Internet at this time. Rather, it is hoped that a general consensus will emerge as to the appropriate approach to these issues, leading eventually to the adoption of ARPA-Internet standards. Distribution of this memo is unlimited.

1. Introduction

This report is a comparison of several data representation standards that are currently in use. The standards, or system type definitions, that will be discussed are the CCITT X.409 recommendation, the NBS Computer Based Message System (CBMS) standard, DARPA Multimedia Mail system, the Courier remote procedure call protocol, and the SUN Remote Procedure Call package.

One purpose of this report is to determine how the CCITT standard, which is gaining wide acceptance internationally, compares with some of the other standards that have been developed in the areas of electronic mail, distributed interprocess communication, and remote procedure call. The CCITT X.409 recommendation, which is entitled "Presentation Transfer Syntax and Notation" is an international standard which is a part of the X.400 series Message Handling Systems (MHS) specifications [1]. It has been adopted by both the NBS and the ISO standards organizations. In addition, some commercial organizations have announced intentions to support a CCITT interface for electronic mail. The NBS Computer Based Message System (CBMS) standard was developed previously and was published as a Federal Information Processing Standard (FIPS Publication 98) in 1983 [3]. The DARPA Multimedia Mail system is an experimental electronic mail system which is in use in the DARPA Internet [2,4,5]. It is used to create and distribute messages that incorporate text, graphics, stored speech, and images and has been implemented on on several very different machines. Courier is the XEROX network systems remote procedure call protocol [7]. The SUN Remote Procedure Call package implements "network pipes" between UNIX machines [6].

2. Background

This section presents a brief overview of the basic terminology and approach of each data representation standard.

2.1. Interprocess Communication Standards

The standards that are oriented towards distributed interprocess communication or remote procedure call, between like machines, generally favor the use of types that map easily into the types defined in the programming language in use on the system. For example, the types defined for the XEROX Courier system resemble the types found in the Mesa programming language. Similarly, the SUN Remote Procedure Call system types resemble the types found in the C programming language. An advantage of a system implemented using like machines is that the external data representation can be defined in such a way that the conversion to and from the local format is minimal.

2.1.1. Courier

The Courier standard data types are used to define the data objects which are transported bi-directionally between system elements that are running the Courier remote procedure call protocol. The "standard representation" of a type is the encoding of the data which is transmitted. The "standard notation" refers to the conventions for the interpretation of the data by higher-level applications. The standard representation of a data object encodes the value of the object, but the type of the object is determined by the software that generates or interprets the representation.

2.1.2. SUN Remote Procedure Call Package

The SUN Remote Procedure Call package includes routines which allow a process on one UNIX machine to consume data produced by a process on another UNIX machine. This is called a "network pipe" and is an extension of the standard UNIX pipe. The "eXternal Data Representation (XDR)" standard defines the routines that are used to encode or "serialize" data for transmission, or to decode or "deserialize" data for local interpretation. The syntax suggests that perhaps it should be called "remote interprocess communication" rather than "remote procedure call".

2.2. Message Standards

The message oriented standards, including DARPA Multimedia Mail, NBS CBMS, and the CCITT X.409 standards, seem to favor more general, highly extensible type definitions. This may have something to do with the expectation that a system will include many different machines, programmed using many different programming languages.

2.2.1. DARPA Multimedia Mail

The DARPA Multimedia Mail system was developed for use in DoD Internet community. The set of data elements used in the Multimedia Message Handling Facility (MMHF) is referred to as its "presentation transfer syntax". The encoding of these data elements varies with the data type being represented. Each begins with a one-octet "element-code". Some data elements are of a pre-determined length. For example, the INTEGER data element occupies five octets, one for the element-code, and four which contain the "value component". Other data elements, however, may vary in length. For example, the TEXT data element, is made up of a one-octet element-code, a three-octet count of the characters to follow, and a variable number of octets, each containing one right-justified seven bit ASCII character. The element-code and the length constitute the "tag component".

A "base data element" is self contained, while a "structured data element" is formed using other data elements. The LIST data element is used to create structures composed of other elements. The tag component of a LIST is made up of a one-octet element-code, a three-octet count of the number of octets to follow, and a two-octet count of the number of elements that follow. The PROPLIST data element is used to create a structure that consists of a set of unordered name-value pairs. The tag component of a PROPLIST is made up of a one-octet element-code, a three-octet count of the number of octets to follow, and a one-octet count of the number of name-value pairs in the PROPLIST. Both the LIST and the PROPLIST elements are followed by an ENDLIST data element.

2.2.2. NBS Computer Based Message System

The NBS Computer Based Message System (CBMS) standard was developed to specify the format of a message at the interface between different computer-based message systems. Each data element consists of a series of "components". The five

possible types of component are the "identifier octet", the "length code", the "qualifier", the "property-list" component, and the "data element contents". Every data element contains an identifier octet and a length code. The identifier octet contains a one-bit flag that signifies whether the data element contains a property-list, and a code identifying the data element and signifying whether it contains a qualifier. In the NBS standard, the property-list is associated with a data element and contains properties such as a "printing-name" or a "comment". The meaning of the qualifier depends on the data element code. The length code indicates the number of octets following, and is between one and three octets in length.

Each data element is inherently a "primitive data element", which contains a basic item of information, or a "constructor data element", which contains one or more data elements. The "field" data element (itself a constructor) uses a qualifier component, which contains a "field identifier" to indicate which specific field is being represented within a message.

2.2.3. CCITT Recommendation X.409

The CCITT recommendation X.409 defines the notation and the representational technique used to specify and to encode the Message Handling System (MHS) protocols. The following is a description of the CCITT approach to encoding type definitions. A data element consists of three components, the "identifier" (type), the "length", and the "contents". An element and its components consist of a sequence of an integral number of octets. An identifier consists of a "class" ("universal", "application-wide", "context-specific", or "private-use"), a "form" ("primitive" or "constructor"), and the "id code". There is a convention defined for both single-octet and multi-octet identifiers. The length specifies the length of the contents in octets, and is itself variable in length. There is also an "indefinite" value defined for the length; this means that no length for the contents is specified, and the contents is terminated with the the "end-of-contents" (EOC) element. In X.409 it is possible to determine whether a data element is a primitive or a constructor from the form part of the identifier. In addition it is possible to "tag" the data by attaching meaning to an id code within the context of a specific application.

3. Implicit Versus Explicit Representation

In both the SUN Remote Procedure Call system and the XEROX Courier system the type definitions of external data are implicit. This means that for a given type of call, or message, the type definitions which is to be used to interpret the data, are agreed upon by the sender and the receiver in advance. In other words, parameters (or message fields) are assumed to be in a predefined order. Each parameter is assumed to be of a predefined type. This means the data cannot be reformated into the local form until it reaches a process that knows about the types of specific parameters. At this point, the conversion can be accomplished using system routines that know how to convert from the external format to the local format. If the system is homogeneous there may be very little conversion required. In addition, no extra overhead of sending the type definitions with the data is incurred.

In the DARPA Multimedia Mail system, the NBS CBMS standard, and the CCITT X.409 recommendation, type definitions are explicit. In this case the type definitions are encoded into the message. There are several advantages to this approach. One advantage is that it allows a low level receiver process in the destination host to convert the data from the standard form to a form appropriate for the local host, as it received. This can increase efficiency if it allows the destination host to avoid passing around data that does not conform to the local word boundaries. Another advantage is that it provides flexibility for future expansion. Since the overall length is a part of the type definition, it allows a host to deal with or ignore data of types that it does not necessarily understand. Since the interpretation of the data is not dependent on its position, message fields (or parameters) can be reordered, or optionally omitted. The disadvantages of this approach are as follows. Assuming that no field could be omitted, the external representation of the message may be longer than it would have been if an implicit representation had been used. In addition, extra time may be consumed by the conversion between external format and local format, since the external format almost certainly will not match the local format for any of the participants.

4. Data Representation Standards Scorecard

The following table is a comparison of the data elements defined for the various standards being discussed. It is provided in order to give a general idea of the types defined for each standard, but it should be noted that the grouping of these types does not indicate one type corresponds exactly to any other. Where it is applicable, the identifier code appears in parantheses following the name of the data element. Under "NUMBER", "S" stands for signed, "U" stands for unsigned, "V" stands for variable, and the number represents the number of bits. For example, "Integer S16" means a "signed 16-bit integer".

Type	CCITT	MMM	NBS	XEROX	Sun
END	End-of-Contents (0)	ENDLIST (11)	End-of-Constructor (1)	--	--
PAD	Null (5)	NOP (0) PAD (1)	No-Op (0) Padding (33)	--	--
RECORD	Set (17) Sequence (16)	PROPLIST (14) LIST (9)	Set (11) Sequence (10) Message (77)	-- Sequence Record	-- Structure
	--	--	--	Array	Fixed Array Counted Array
	"Choice" "Any"	--	--	Choice	Discriminated-Union
	"Tagged"	"name"	Field (76) Unique-ID(9)	--	--
	--	SHARE-TAG (12) SHARE-REF (13)	--	--	--
	--	--	Compressed (70)	--	--
	--	ENCRYPT (14)	Encrypted (71)	--	--

Type	CCITT	MMM	NBS	XEROX	Sun
BOOLEAN	Boolean(1)	BOOLEAN(2)	Boolean(8)	Boolean	Boolean
NUMBER	Integer(2) SV	EPI (5) SV INDEX (3) U16 INTEGER(4) S32	Integer(32) SV	Integer S16 Cardinal U16 Unspecified 16 Long Int S32 Long Card U32	Integer S32 Unsigned Int U32 Enumeration 32 Hyper Integer S64 Uns Hyper Int U64 Double Prec 64
	--	FLOAT (15) 64	--	--	Float Pt 32
BIT-STRING	Bit String (3)	BITSTR(6)	Bit-String (67)	--	--
	Octet-String(4)	--	--	--	Opaque
STRING	IA5 (22)	TEXT (8)	ASCII-String (2)	String	Counted-Byte String
		NAME (7)			
	Numeric (18)				
	Printable (19)				
	T.61 (20)				
	Videotex (21)				

Type	CCITT	MMM	NBS	XEROX	Sun
OTHER	UTC Time (23) Gen Time (24)	--	Date (40)	--	--
	--	--	Property-List (36)	--	--
	--	--	Property(69)	--	--
	--	--	--	Procedure	--
	--	--	Vendor-Defined (127) Extension (126)	--	--

5. Conclusions

Of the standards discussed in this survey, the CCITT approach (X.409) has already gained wide acceptance. For a system that will include a number of dissimilar hosts, as might be the case for an Internet application, a standard that employs explicit representation, such as the CCITT X.409, would probably work well. Using the CCITT X.409 standard it is possible to construct most of the data elements that are specified for the other standards, with the possible exception of the "floating point" type. However, some of the flexibility that has been built into this standard, such as the "private-use class" may lead to ambiguity and a lack of coordination between implementors at different sites. If a standard such as the CCITT were to be used in an Internet experiment a fully defined (but large) subset would probably have to be selected.

6. References

[1] "Message Handling Systems: Presentation Transfer Syntax and Notation", Recommendation X.409, Document AP VIII-66-E, International Telegraph and Telephone Consultative Committee (CCITT), Malaga-Torremolinos, June, 1984.

[2] J. Garcia-Luna, A. Poggio, and D. Elliot, "Research into Multimedia Message System Architecture", SRI International, February, 1984.

[3] "Specification for Message Format for Computer Based Message Systems", FIPS Pub 98 (also published as RFC 841), National Bureau of Standards, January, 1983.

[4] J. Postel, "Internet Multimedia Mail Transfer Protocol", USC Information Sciences Institute, MMM-11 (RFC-759 revised), March, 1982.

[5] J. Postel, "Internet Multimedia Mail Document Format", USC Information Sciences Institute, MMM-12 (RFC-767 revised), March, 1982.

[6] "Extended Data Representation Reference Manual", SUN Microsystems, September, 1984.

[7] "Courier: The Remote Procedure Call Protocol", XSIS-038112, XEROX Corporation, December, 1981.